167657 - Controle para Automação
Curso de Graduação em Engenharia de Controle e Automação
Departamento de Engenharia Elétrica
Universidade de Brasília

Linguagens e Autômatos

Geovany A. Borges
gaborges@ene.unb.br

Conceitos básicos de linguagem para SEDs

- Linguagens em SEDs: usadas para representar seqüências de eventos
  - Formadas a partir de um alfabeto representado pelo conjunto de eventos E.
  - Seqüência de eventos do alfabeto forma uma palavra (ou *string*)
  - Algumas interpretações do formalismo:

"Pode-se construir um sistema que fala uma determinada língua?"

"Que língua este sistema fala?"

"Pode-se alterar o dialeto?"

- Autômato: modelo que permite representar a "linguagem" de um SED.

Conceitos básicos de linguagem para SEDs

- Definição de linguagem: conjunto de strings de tamanho finito formado a partir de eventos de E.

  - String formado por nenhum evento: ε
  - $\varepsilon \notin E$ mas também $\varepsilon \notin \emptyset$.
  - $|s| \rightarrow$ Comprimento do string s, que é tomado pelo número de eventos que o compõem, mesmo que estejam repetidos.
  - Por convenção, $|\varepsilon| = 0$.
  - Uma linguagem pode ou não conter ε.

  Exemplos: sendo $E = \{a, c, x\}$, tem-se como exemplos de linguagens

  - $L_1 = \{\varepsilon, a, ccac\}$
  - $L_2 = \{$conjunto de todos strings de tamanho 3 iniciando com o evento $a\}$. Verifique que $|L_2| = 9$.

Conceitos básicos de linguagem para SEDs

- 2^E : conjunto potência de E, que corresponde ao conjunto de todos os subconjuntos de E.

  Exemplo: sendo $E = \{a, c, x\}$

  $$2^E = \{\{a\}, \{c\}, \{x\}, \{a, c\}, \{a, x\}, \{c, x\}, \{a, c, x\}\}$$

- Concatenação: abc é resultado da concatenação dos strings a e bc.

  ε é o operador identidade da concatenação: $\varepsilon s = s\varepsilon = s$ para qualquer string s.

Conceitos básicos de linguagem para SEDs

- Fechamento de Kleene: conjunto E^* formado por todos os strings formados a partir de elementos de E, mas que inclui ε.

  Sendo $E = \{a, b\}$ então $E^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, \cdots\}$

  Pode-se verificar que

  - $|E^*| = \infty$
  - Uma linguagem contruída a partir de E é um subconjunto de E^*

- Sendo $s = abcd$, então define-se

  - a é prefixo de s e v é sufixo
  - $\varepsilon, b, c, bc, abc, bcd$ e $abcd$ são substrings de s
  - ε e s são prefixos, sufixos e prefixos de s.

Operações com linguagens

Pode-se perceber que linguagens são na verdade conjuntos. Portanto, todas as operações de teoria dos conjuntos se aplicam a linguagens.

- Concatenação: seja $L_a, L_b \subseteq E^*$, então

$$L_a L_b := \{s \in E^* : (s = s_a s_b) \text{ e } (s_a = L_a) \text{ e } (s_b = L_b)\}$$

- Fechamento de prefixo: seja $L \subseteq E^*$

$$\bar{L} := \{s \in E^* : \exists t \in E^* \text{ tal que } (st \in L)\}$$

que é o conjunto formado por todos os prefixos de todos os strings de L.

Linguagem de prefixo fechado: $L = \bar{L}$.

Se $L \neq \oslash$ então $\varepsilon \in \bar{L}$.

Operações com linguagens

- Fechamento de Kleene de uma linguagem: seja $L \subseteq E^*$

$$L^* = \{\varepsilon\} \cup L \cup LL \cup LLL \cup \cdots$$

Deve ser observado que $(L^*)^* = L^*$.

Operações com linguagens

Exercício 1: *Sendo* $E = \{a, b, g\}$ *e as linguagens* $L_1 = \{\varepsilon, a, abb\}$ *e* $L_4 = \{g\}$*, responda:*

I. Estas linguagens são de prefixo fechado?

II. Determine L_1L_4*,* $\bar{L}_4$ *e* L_1^* $\Diamond$

Exercício 2: *As afirmativas abaixo são verdadeiras ou falsas?*

I. $baa \in \{a\}^*\{b\}^*\{a\}^*\{b\}^*$

II. $\{b\}^*\{a\}^* \cap \{a\}^*\{b\}^* = \{a\}^* \cup \{b\}^*$ $\Diamond$

Operações com linguagens

Exercício 3: *Considere um buffer de uma porta de comunicação que pode comportar no máximo 2 mensagens que ainda não foram processadas. Considere que o buffer esteja inicialmente vazio, que* $E = \{r, p\}$ *com* $r =$ *mensagem recebida e* $p =$ *mensagem processada, e que apenas uma mensagem pode ser processada por vez. Determine a linguagem deste sistema.* $\Diamond$

Exercício 4: *Considere* $E = \{c, i, f, d\}$ *eventos de uma máquina de manufatura:* $c =$ *chegada de material,* $i =$ *início do procedimento de corte,* $f =$ *conclusão do procedimento de corte e* $d =$ *despacho do material. Considere também que*

I. não existe limite para o número de materias à espera de processamento

II. os materiais são despachados logo após a conclusão do procedimento de corte

III. apenas um material pode estar em procedimento de corte por vez

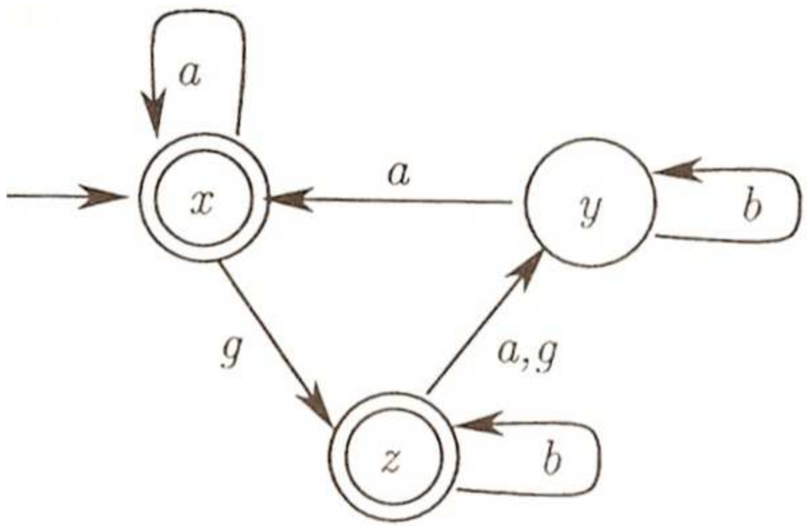
Dê exemplos de palavras da linguagem deste processo. $\Diamond$

Autômato

- Ferramenta de representação compacta de linguagens em SEDs.

- Por quê? Tente usar a notação de conjunto para representar uma linguagem tal que $|L| = \infty$.

- Definição informal:



Autômato

- Autômato determinístico: dado pela sextupla

$$G = (X, E, f, \Gamma, x_0, X_m)$$

  - X: conjunto de estados;
  - E: conjunto finito de eventos associados às transições de G;
  - $f : X \times E \to X$: função de transição, que pode ser parcial. Algumas vezes usa-se a forma recursiva:

$$f(x, ab) = f(f(x, a), b) \text{ para } a \in E^* \text{ e } b \in E.$$

  - $\Gamma : X \to 2^E$: função de eventos ativos (conjunto de eventos factíveis para cada estado). Informação redundante pois pode ser obtida a partir de f;
  - $X_m \subseteq X$: conjunto de estados marcados.

- Como fica o modelo do autômato da transparência anterior?

Linguagens representadas pelo autômato

- Linguagem representada pelo autômato G:

$$\mathcal{L}(G) := \{s \in E^* : f(x_0, s) \text{ está definido}\}$$

- Linguagem marcada pelo autômato G:

$$\mathcal{L}_m(G) := \{s \in \mathcal{L}(G) : f(x_0, s) \in X_m\}$$

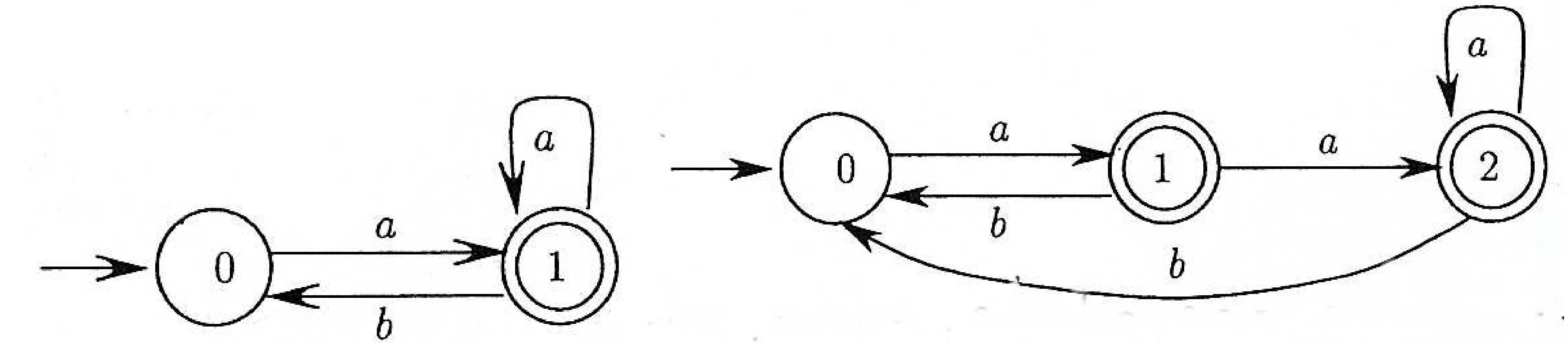
- Lista de Exercícios: 2.5, 2.6, 2.7, 2.8, 2.9, 2.10, 2.11, 2.12 e 2.13.

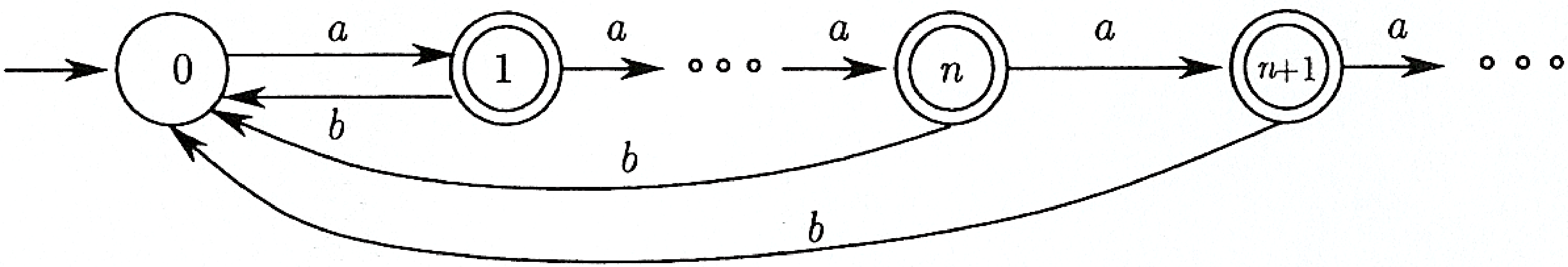
Linguagens representadas pelo autômato

- Exercício: Modelamento de uma máquina de refrigerantes
  - $E = \{10, 20\}$
  - $X = \{x_0, x_{10}, x_{20}, x_{30}, x_{40}\}$ em que $X_m = \{x_{40}\}$
  - Limitações do modelo?
  - Solução usando autômato evento/saída.

Equivalência de Autômatos

- G_1 e G_2 são ditos equivalentes se $\mathcal{L}(G_1) = \mathcal{L}(G_2)$ e $\mathcal{L}_m(G_1) = \mathcal{L}_m(G_2)$.
- Exemplo de autômatos equivalentes:





Bloqueio

- Se um autômato atingir um estado x para o qual $\Gamma(x) = \oslash$ e $x \notin X_m$, então tem-se uma situação de *deadlock*.

- Se existir um sub-conjunto de estados não marcados de tal forma que, se um deles for alcançado, não existe evento que faça o autômato sair do sub-conjunto de estados, então tem-se uma situação de *livelock*.

- Um autômato é dito *bloqueável* se

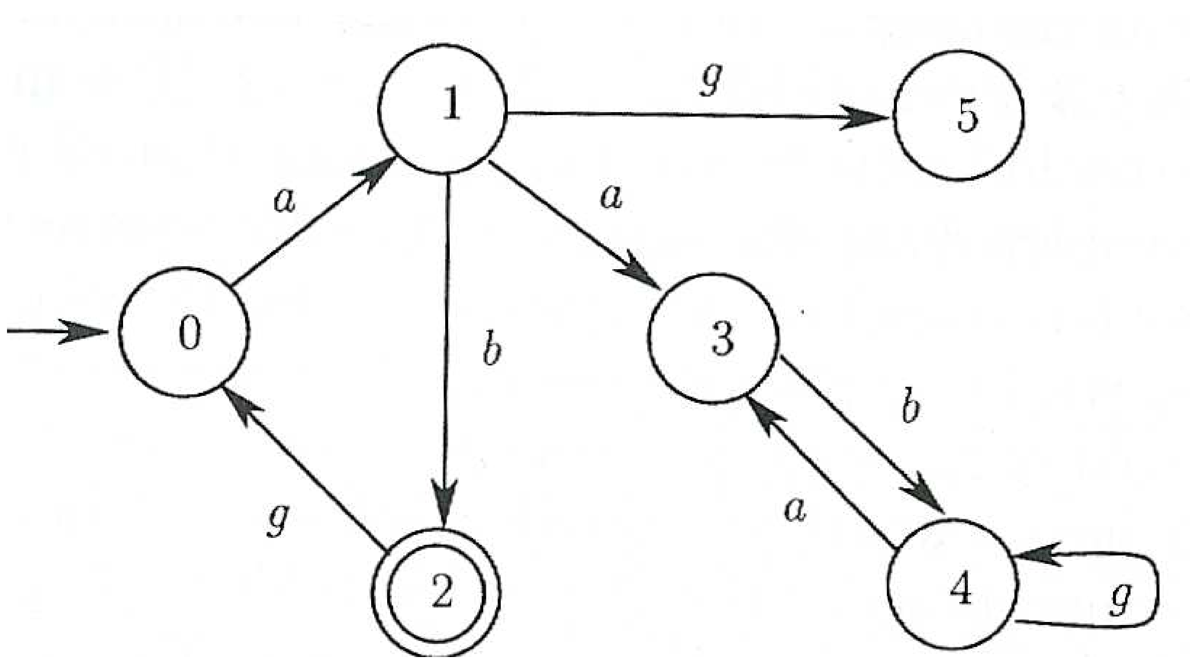
$$\overline{\mathcal{L}_m(G)} \subset \mathcal{L}(G)$$

e *não-bloqueável* se

$$\overline{\mathcal{L}_m(G)} = \mathcal{L}(G)$$

Bloqueio

- Exemplo: o que dizer a respeito do string ag? O que dizer dos estados 3 e 4?



Autômato não-determinístico

- Característica principal de autômatos não-determinísticos: dois ou mais estados podem estar ativos simultaneamente.

- Situações de uso:
  - Quando não se sabe ao certo qual estado o sistema se encontra após ocorrido um evento (problema de observabilidade)
  - O símbolo ε pode vazer parte explícita do conjunto de eventos

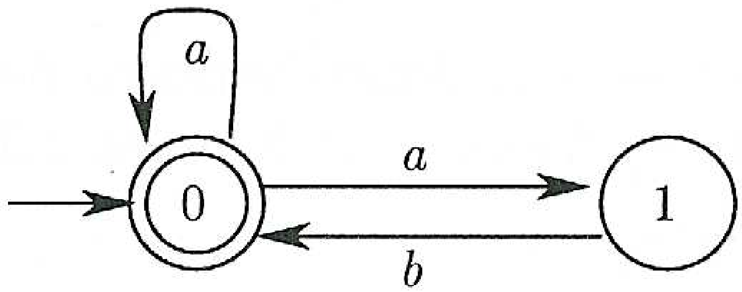
- Definição: um autômato não-determinístico é dado pela sextupla

$$G_{nd} = (X, E \cup \{\varepsilon\}, f_{nd}, \Gamma, x_0, X_m)$$

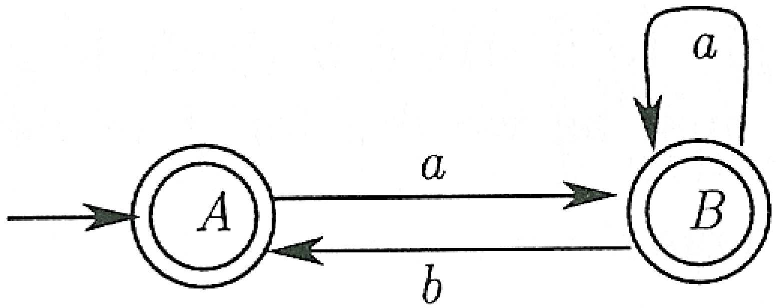
com $f_{nd} : X \times E \cup \{\varepsilon\} \to 2^X$ e $x_0 \subseteq X$ podendo $|x_0| \geq 1$.

Autômato não-determinístico

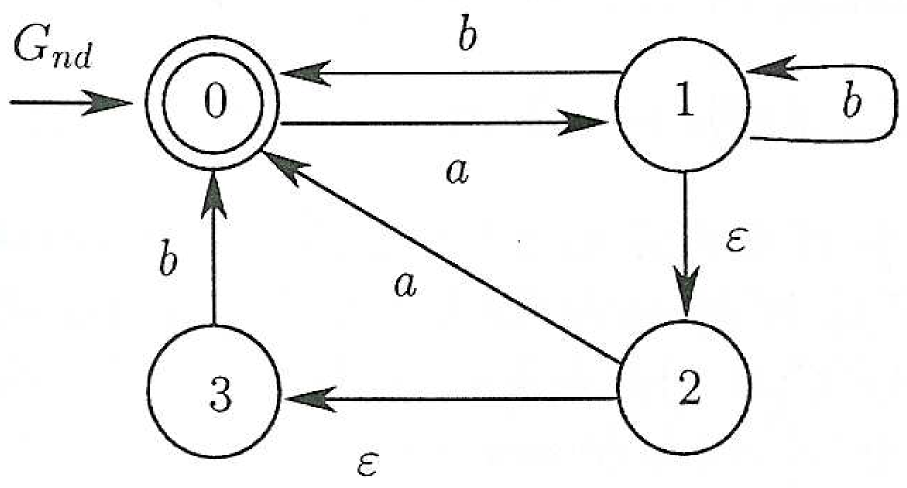
- Exemplo: determine $\mathcal{L}(G_{nd})$ e $\mathcal{L}_m(G_{nd})$ para o autômato abaixo:



Um equivalente determinístico pode ser o autômato seguinte:



Autômato observador

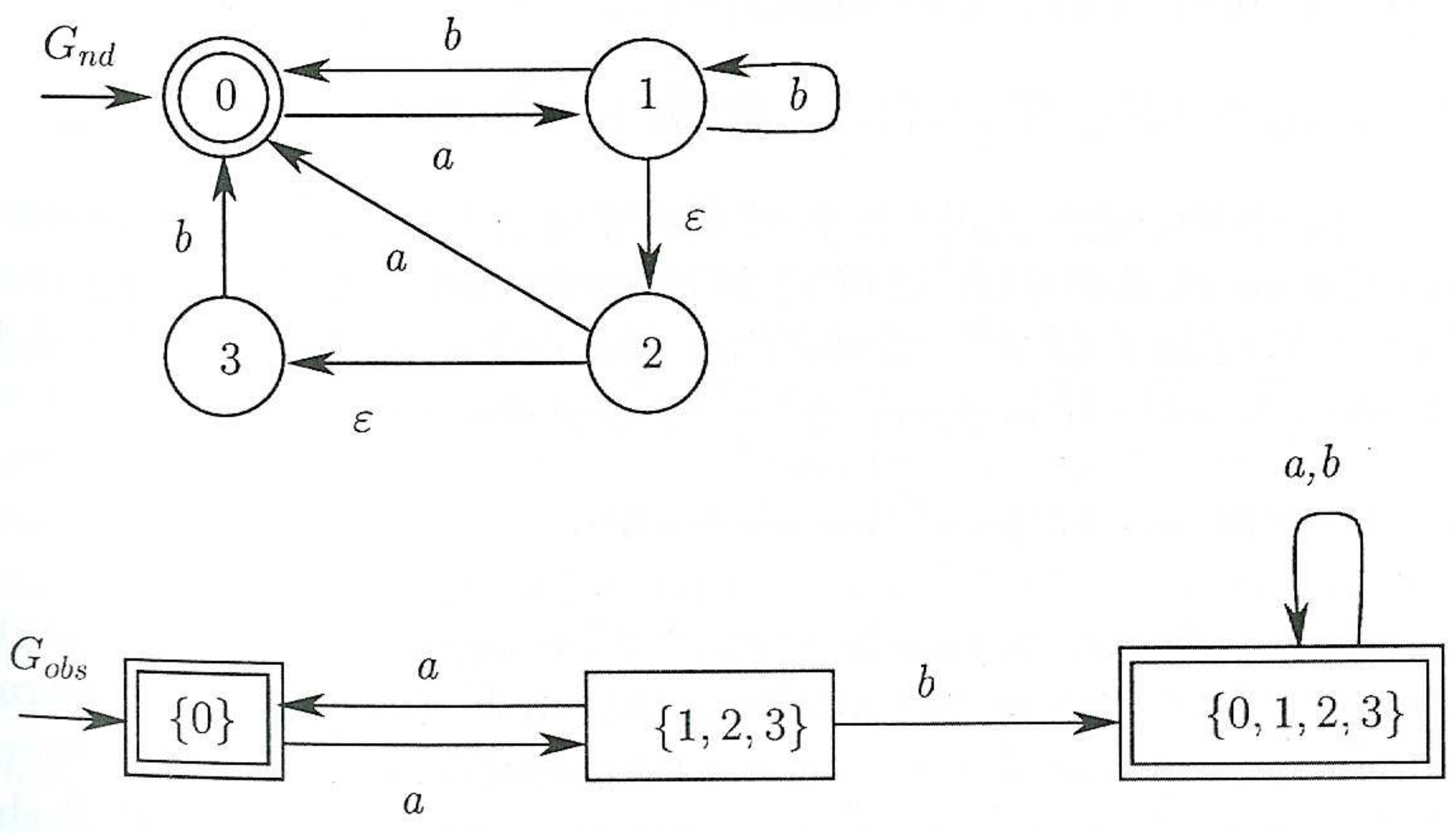


- Definição: sendo G_{nd} um autômato não-determínstico.
  - G_{obs} é um autômato determinístico equivalente a G_{nd};
  - G_{obs} desempenha o mesmo papel de um observador de estados

$$G_{nd} = (X, E \cup \{\varepsilon\}, f_{nd}, \Gamma, x_0, X_m) \Longrightarrow G_{obs} = (X_{obs}, E, f_{obs}, \Gamma, x_{0,obs}, X_{m,obs})$$

Autômato observador

- Exemplo de um autômato observador:



Autômato observador

- Algoritmo de obtenção do autômato observador (interpretação do livro-texto):

  I. Prepare uma tabela para representar candidatos $x'_{obs} \in 2^X$ versus eventos $e \in E$. Deixe algumas linhas em branco para novos estados. Preencha as primeiras linhas apenas com os estados do conjunto X.

  II. Dermine novos estados candidatos $x'_{obs} = f_{nd}(x, e\varepsilon^n)$, com $n = 0, 1, 2, \ldots$, em função do evento e. Observe que os novos estados x'_{obs} são conjuntos.

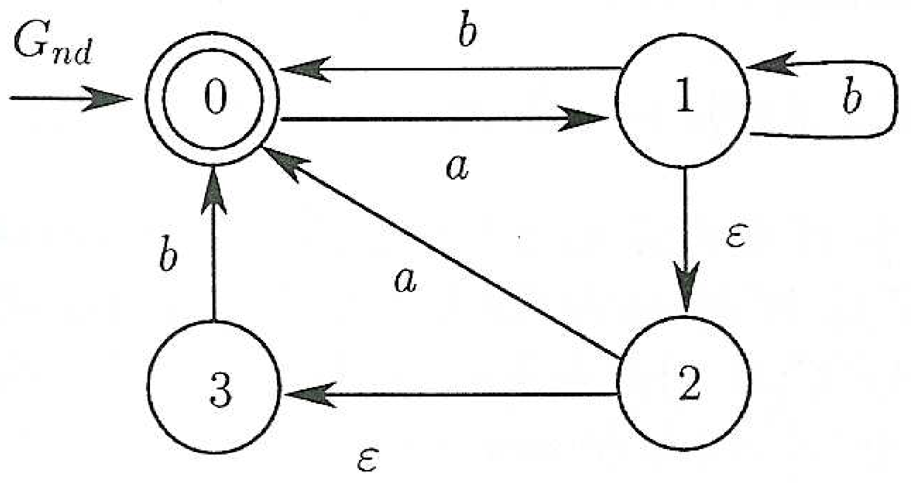
  III. Partindo de cada x'_{obs} obtido de x_0, gere novos estados candidatos buscando fazer uso, na medida do possível, da seguinte propriedade:

$$f_{nd}(x'_{obs}, e\varepsilon^n) = \bigcup_{x \in x'_{obs}} f_{nd}(x, e\varepsilon^n) \qquad \text{com } n = 0, 1, 2, \ldots.$$

  IV. Partindo de x_0, determine $x_{0,obs} = f_{nd}(x_0, \varepsilon^n)$. Monte o autômato mantendo apenas os estados candidatos x_{obs} dentre os x'_{obs} alcancáveis a partir de $x_{0,obs}$. Será marcado todo estado x_{obs} que possuir entre seus elementos um estado marcado do conjunto inicial X_m.

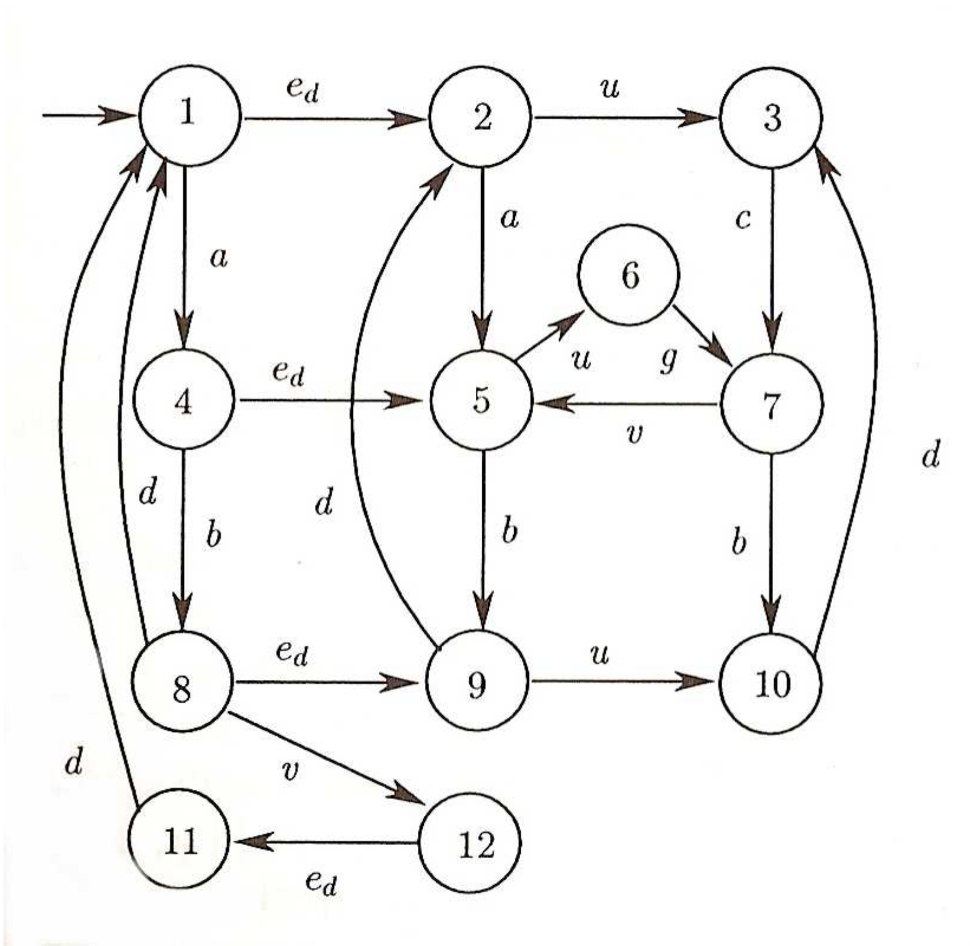
Autômato observador

- Exemplo: obtenha um autômato observador para o G_{nd} abaixo.



Autômato observador

- Exercício: obtenha um autômato observador para o G_{nd} abaixo (Fig. 2.24, pp113).



Autômato observador

- Exercício: problema 2.29 (pp128). Substituir a por ε.

2.29 Consider the automaton H depicted in Fig. 2.36. The set of events is $E = \{a, b, c, d\}$ and the set of observable events is $E_o = \{b, c, d\}$. Build H_{obs}, the observer of H. Compare the blocking properties of H with those of H_{obs}.

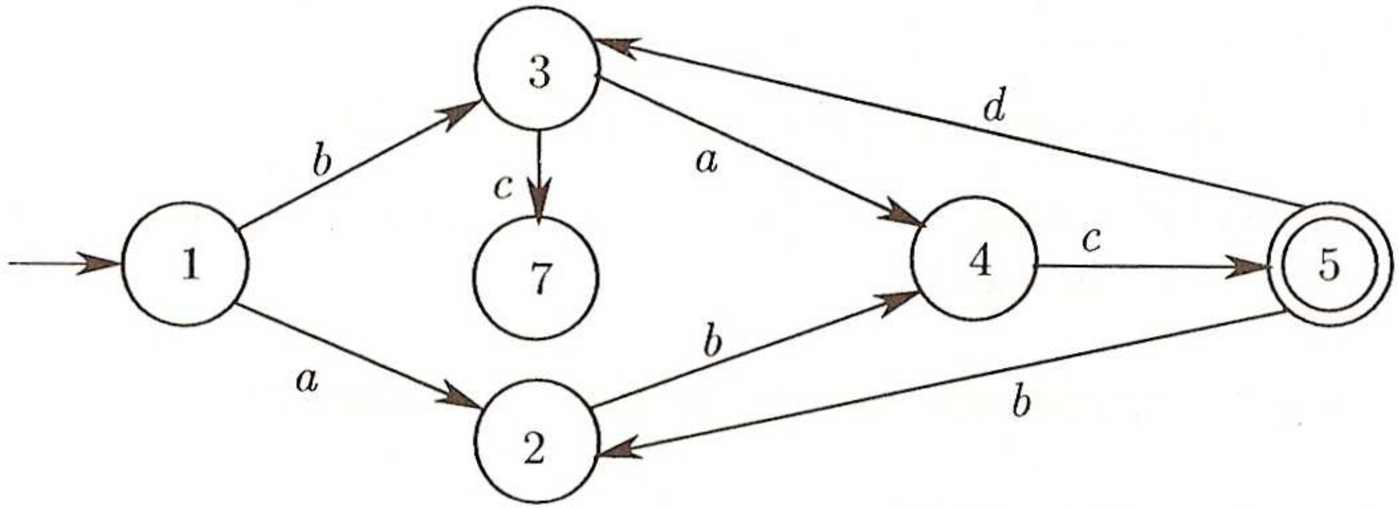
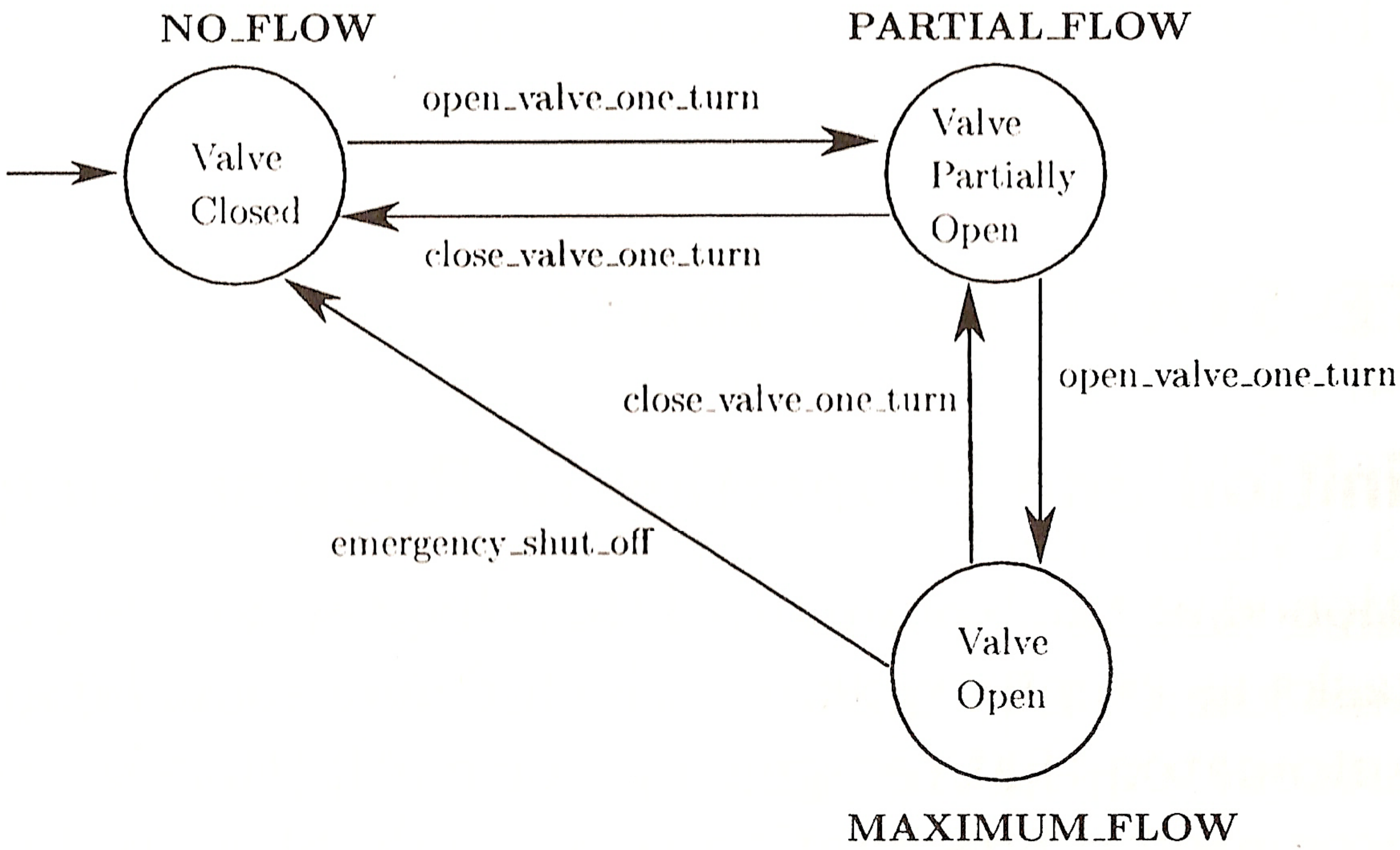


Figure 2.36: Problem 2.29.

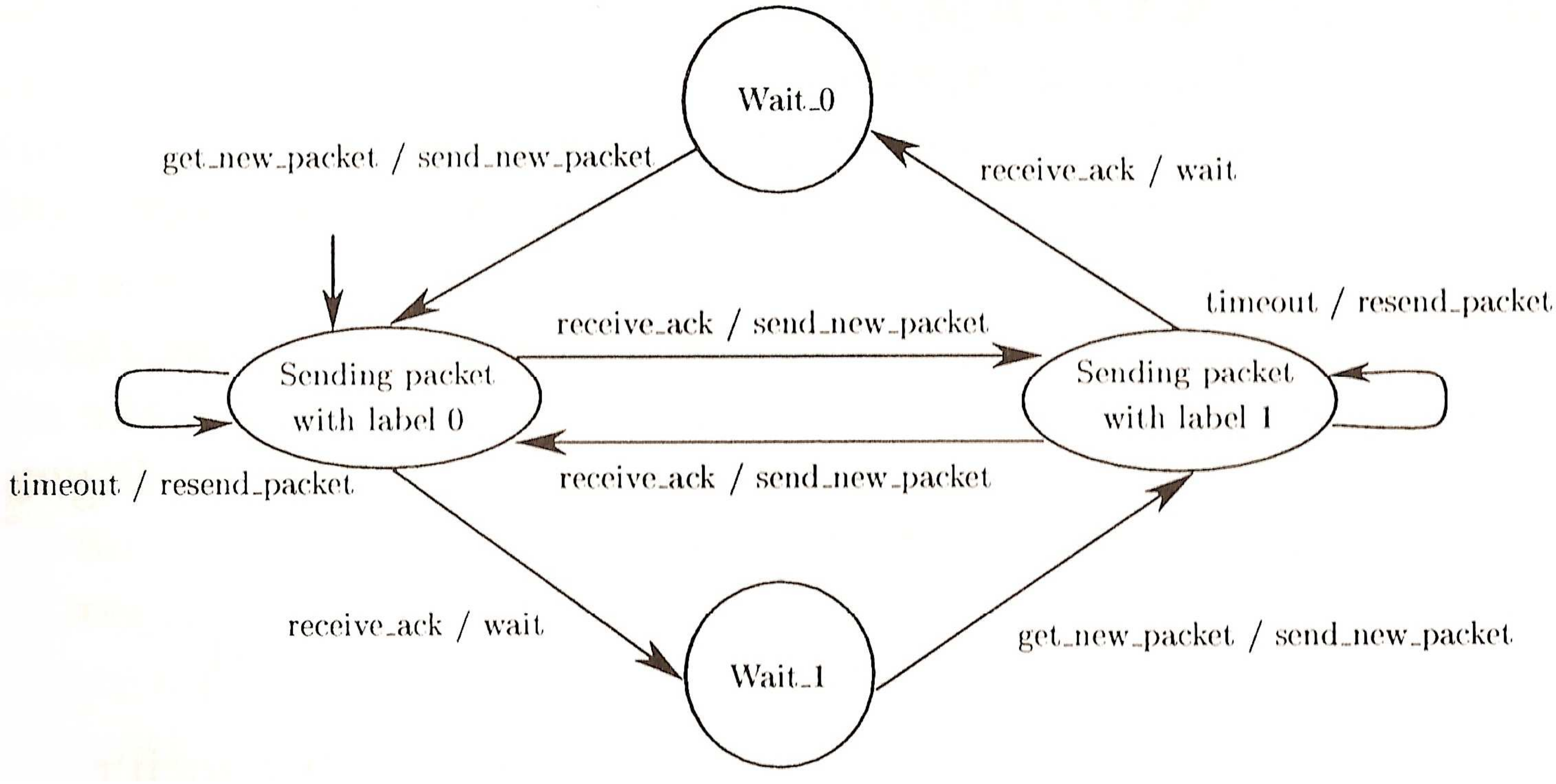
Autômato com entradas e saídas

- Autômato de Moore: autômato com saídas associadas aos estados.



Autômato com entradas e saídas

- Autômato de Mealy: autômato com arcos rotulador por evento de endrada/ evento de saída.



Estimação de estados e diagnóstico

- O autômato observador nada mais é do que um estimador de estados para sistemas em que nem todos os eventos são detectáveis. Se todos os eventos são mensuráveis, então $G_{obs} = G$.

- Uma interpretação para ε em um autômato não-determinístico: eventos não observáveis (não mensuráveis, ou não detectáveis ou simplesmente não modelados).

- Partição de E:

$$E = E_o \cup E_u$$

  com E_o representando os eventos observáveis e E_u os eventos não observáveis.

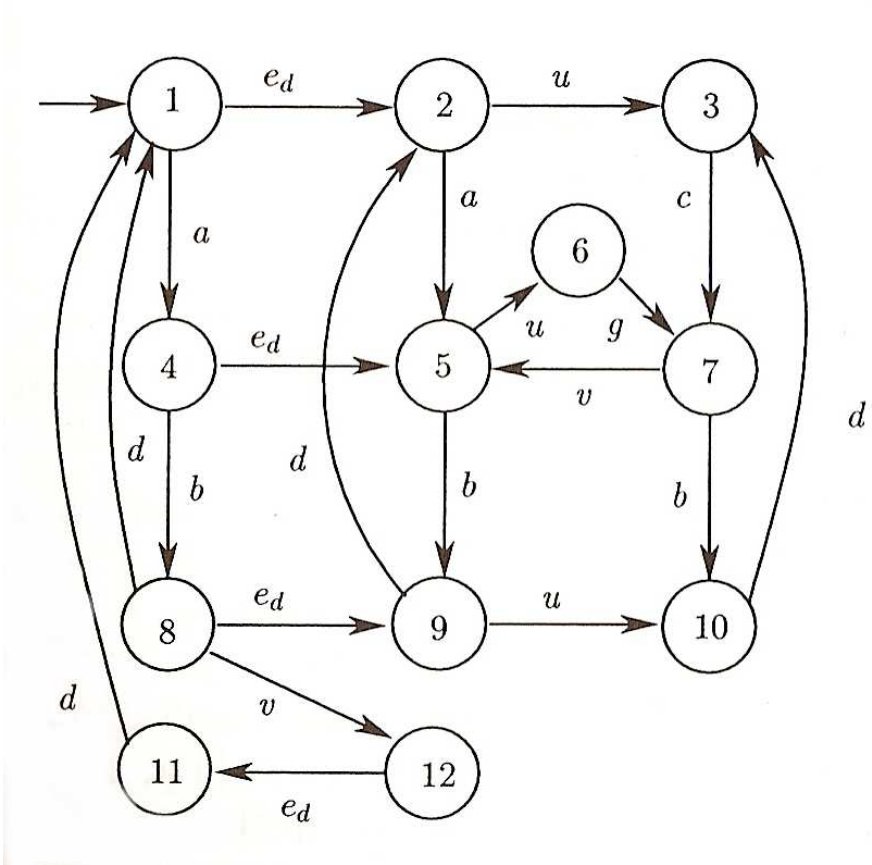
- Problema do diagnóstico: por meio de obervações de E_o, determinar, com certeza ou não, a ocorrência de eventos de E_u. Será considerado apenas o caso de diagnosticar a ocorrência um único evento $e_d \in E_u$.

Estimação de estados e diagnóstico

- Algoritmo de obtenção do autômato diagnosticador G_{diag} para o evento e_d: mesmos procedimentos usados para a construção do autômato observador, mas com as alterações seguintes:
  - Alteração 1: no passo 2 do algoritmo para obter G_{obs}
    - Associe o rótulo N para estados que podem ser alcançados a partir de x_0 por strings de $[E_u \backslash \{e_d\}]^*$;
    - Associe o rótulo Y para estados que podem ser alcançados a partir de x_0 por strings de $[E]^*$ que contenham pelo menos uma ocorrência de e_d;
    - Se existir um estado x que pode ser alcançado a partir de x_0 por strings de $[E]^*$ *com* e *sem* a ocorrência de e_d, então cria-se dois estados candidatos de G_{diag}: xN e xY.
  - Alteração 2:
    - Aplique as regras do algoritmo original para obtenção de G_{obs}, mas procurando aplicar as regras da Alteração 1 acima.
    - Propague Y: qualquer novo estado derivado do estado zY deve conter também o rótulo Y, indicando que e_d já ocorreu desde a condição inicial.

Estimação de estados e diagnóstico

- Exercício: construir G_{diag} para o G_{nd} abaixo considerando $E_u = \{e_d, u, v\}$ (Fig. 2.24).



Referências